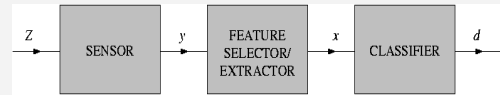


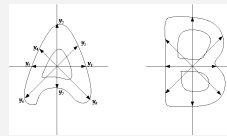
Dimensionality reduction

Josef Kittler

Centre for Vision, Speech and Signal Processing, University of Surrey
email: J.Kittler@surrey.ac.uk



Pattern representation



Pattern recognition problem

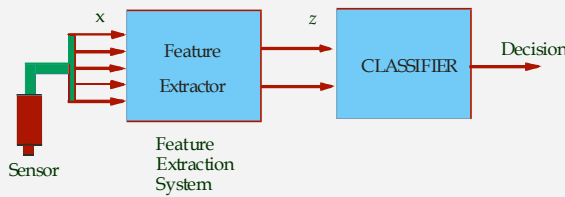
m – number of classes

x - pattern vector

$P(\omega_i)$ -priori probability of class ω_i

$p(x|\omega_i)$ -measurement distribution of patterns in class ω_i

Number of measurements
May be very large!



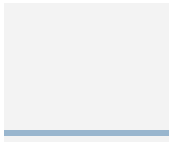
$$\mathbf{x} = [x_1, x_2, \dots, x_5]^T \text{ PATTERN REPRESENTATION VECTOR}$$

$$\mathbf{z} = [z_1, z_2]^T \text{ Feature vector}$$

- Kernel methods
 - Kernel methods: an overview
 - Kernel PCA and Kernel Discriminant Analysis
- Multiple Kernel Discriminant Analysis
 - MKL: motivation
 - ℓ_p regularised Multiple Kernel Discriminant Analysis
 - The effect of regularisation norm
 - MKL and feature space denoising
- Summary

What is PCA

- Principal component analysis (PCA): an orthogonal basis transformation
- Transform correlated variables into uncorrelated ones (principal components)
- Can be used for dimensionality reduction
- Retains as much variance as possible when reducing dimensionality



How PCA works

- Given m centred vectors: $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$
 - X : $D \times N$ data matrix
- Eigen decomposition of $D \times D$ covariance matrix $C = XX^T$

$$CV - V\Lambda = 0 \quad (1)$$
 - Diagonal matrix Λ : eigenvalues
 - $V = (\mathbf{v}_1, \mathbf{v}_2, \dots)$: eigenvectors (Principal Components)
- Data can now be projected onto orthogonal bases V
- Projecting only onto $d < D$ leading eigenvectors \Rightarrow dimensionality reduction with minimum variance loss

Mapping data to low dimensional feature space

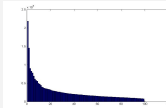
- Given $V = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d]$, the d -dimensional feature vector in the PCA space is given by

$$\mathbf{z} = V^T \mathbf{x}$$

- dimensionality d chosen to retain a certain fraction of variance

- $d < D$
- $d \leq N$

Distribution of eigenvalues



Example of eigenvectors



Kernelising PCA

- Premultiplying eq (1) by data matrix X^T gives

$$X^T X X^T V - X^T V \Lambda = 0 \quad (2)$$
- an eigenvalue problem with eigenvectors $U = X^T V$ and the same eigenvalues
- matrix $X^T X$ is $N \times N$
- if $N \ll D$, then solving eq (2) is more efficient
- matrix V can be obtained as $V = X U \Lambda^{-1}$
- matrix $K = X^T X$ is referred to as kernel matrix
- its element $k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ measures "similarity" of vectors $\mathbf{x}_i, \mathbf{x}_j$
- $k_{ij} = \mathbf{x}_i^T \mathbf{x}_j$

Kernelising PCA

- the kernel formulation of PCA offers a scope for using different notions of similarity
- for instance, $k_{ij} = \exp\{-\|\mathbf{x}_i - \mathbf{x}_j\|^2 \sigma^{-2}\}$
- other notions of similarity remap \mathbf{x} in a nonlinear way into $\phi(\mathbf{x})$
- $\phi(\mathbf{x})$ may be of infinite dimensionality
- $\phi(\mathbf{x})$ is defined implicitly, is unknown, but satisfies $k_{ij} = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$
- working with K is like working with data matrix $X = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_N)]$
- We have kernelised PCA

9

Kernelising PCA

- Note
 - $V = XU\Lambda^{-1}$ is not explicitly available: U and Λ are, but X is not
- However... we are interested in projection onto basis V , not the basis itself
- Projection onto V : $X^T V = X^T XU\Lambda^{-1} = KU\Lambda^{-1}$
- All K and U and Λ are available
- Λ purely rescales the data and can be omitted

10

Kernel Discriminant Analysis

- Kernel Fisher discriminant analysis: another supervised learning technique
- Focusing on discrimination, rather than faithful representation
- Seeking the projection \mathbf{w} maximising Fisher criterion

$$\max_{\mathbf{w}} \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T (S_T + \lambda I) \mathbf{w}} \quad (3)$$

- S_B and S_T : between class and total scatters
- λ : regularisation parameter
- The total scatter matrix equals mixture covariance matrix $S_T = XX^T$
- Between class scatter S_B can be expressed as $S_B = X\Delta X^T$
- Block diagonal matrix Δ contains a constant in block i , proportional to the number of samples from class i

11

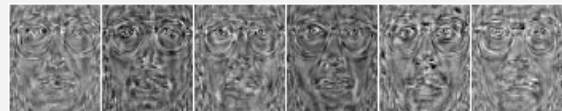
KDA

- Expressing $\mathbf{w} = X\alpha$, and substituting for S_B , we can kernelise Fisher criterion as

$$\max_{\alpha} \frac{\alpha^T X^T X \Delta X^T X \alpha}{\alpha^T X^T (XX^T + \lambda I) X \alpha} = \max_{\alpha} \frac{\alpha^T K \Delta K \alpha}{\alpha^T (K^2 + \lambda K) \alpha} \quad (4)$$

- The maximisation leads to an eigenvalue problem

$$\Delta K \alpha - \kappa (K + \lambda I) \alpha = 0 \quad (5)$$



Spectral regression KDA

- Instead of solving the eigenvalue problem, eq (5) can be solved as a linear system

$$\begin{aligned} \Delta \mathbf{u} &= \eta \mathbf{u} \\ (K + \lambda I) \alpha &= \mathbf{u} \end{aligned} \quad (6)$$

- \mathbf{u} is an eigenvector of matrix Δ , η its eigenvalue
- Owing to the structure of Δ , \mathbf{u} can be found by Gram-Schmidt orthogonalisation
- Expressing $K + \lambda I$ using Cholesky decomposition as $K + \lambda I = R^T R$ where R is an upper triangular matrix, the second linear problem in (6) can be solved as

$$(K + \lambda I) \alpha = \mathbf{u} \Leftrightarrow \begin{cases} R^T \mathbf{v} = \mathbf{u} \\ R \alpha = \mathbf{v} \end{cases} \quad (7)$$

13

Multi Kernel Discriminant Analysis: Motivation

- Ideal case: learn kernel function (notion of similarity) from data
- If that is hard, can we learn a good combination of given kernel matrices: the multiple kernel learning problem
- Given n $N \times N$ kernel matrices, K_1, \dots, K_n
- Most MKL formulations consider linear combination:

$$K = \sum_{j=1}^n \beta_j K_j, \quad \beta_j \geq 0 \quad (6)$$

- Goal of MKL: learn the "optimal" weights $\beta \in \mathbb{R}^n$

14

MKDA optimisation

- Consider linear combination: $\mathcal{K} = \{K = \sum_{i=1}^n \beta_i K_i : \beta \geq \mathbf{0}\}$
- β must be regularised in order for (7) to be meaningful
- Consider a general ℓ_p regularisation for any $p \geq 1$: $\mathcal{K} = \{K = \sum_{i=1}^n \beta_i K_i : \beta \geq \mathbf{0}, \|\beta\|_p \leq 1\}$
- The ℓ_p MKDA problem becomes:

$$\begin{aligned} \Delta \sum_{i=1}^n \beta_i K_i \alpha - \kappa (\sum_{i=1}^n \beta_i K_i + \lambda I) \alpha &= 0 \\ \text{s.t. } \beta &\geq \mathbf{0}, \quad \|\beta\|_p \leq 1 \end{aligned} \quad (7)$$

- Solve using Semi-Infinite Programming
- What norm ℓ_p to use?

15

Regularisation norm effect

- Pascal VOC 2008 development set:
 - 20 object classes \Rightarrow 20 binary problems
 - Mean average precision (MAP) as performance metric

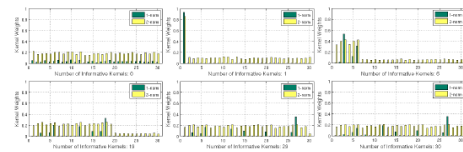


Figure : Learnt kernel weights with various kernel mixture.

- ℓ_1 gives sparse solution and ℓ_2 non-sparse
- A hypothesis: when most kernels are informative sparsity is a bad thing and vice versa

16

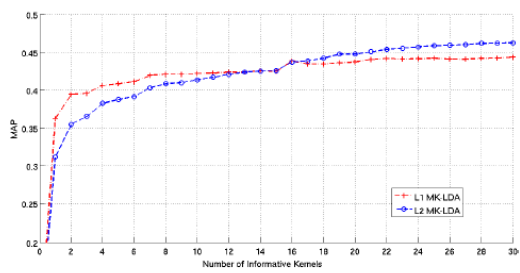


Figure : MAP vs. number of informative kernels

17

■ Pascal VOC 2007

- We have seen the behaviour of ℓ_1 and ℓ_2 MKDAs
- A principle for selecting regularisation norm:
 - High intrinsic sparsity in base kernels: use small norm
 - Low intrinsic sparsity: use large norm
- But how do we know the intrinsic sparsity?
- Simple idea: try various norms, choose the best on validation
- Reduce sparsity by kernel denoising

18

- Basic approach: remove certain percentage of kernel variance
- Use kernel PCA for dimensionality reduction (denoising) in feature space
- Questions to be answered:
 - Can denoising improve single kernel performance?
 - Can denoising improve MKL performance?
 - How does MKL behaviour differ on original kernels and denoised kernels?

19

- PASCAL VOC07 dataset, 20 objects, 33 kernels

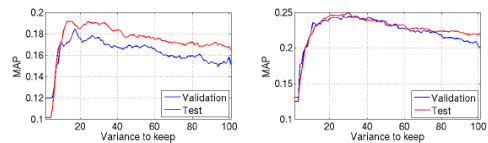


Figure : AP vs. variance kept in kernel PCA. Two kernels as examples.

- Choosing denoising level using a validation set \Rightarrow better single kernel performance (compared to original kernel) for "dining table" and "cat"

20

Denoising: MKL performance

Table : Comparing ℓ_p MK-FDA and fixed norm MK-FDAs

	ℓ_1 MK-FDA	ℓ_2 MK-FDA	ℓ_∞ MK-FDA	ℓ_p MK-FDA
original kernels	54.85	54.79	54.64	55.61
denoised kernels	54.26	56.06	55.82	56.17

- In general, denoised kernels are better than original ones
- ℓ_p is better than fixed norm, on both original and denoised
- Advantage of ℓ_p is much smaller with denoised kernels.

21

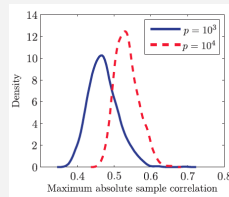
Summary

- Regularisation norm plays an important role in MKL
- ℓ_p MKDA allows to learn intrinsic sparsity of base kernels \Rightarrow better performance than fixed norm MKL
- Feature space denoising is important for KDA and MKDA
 - Denoising improves both single kernel and MKL performance
 - Linear kernel combination cannot take care of feature space denoising automatically

22

Feature selection

- Spurious correlation problem
- Ex.: measuring max pairwise correlation, 50 sample sets, 1000 simulations (Fan 2010)
- Need to impose sparsity (work with a subset of original measurements)



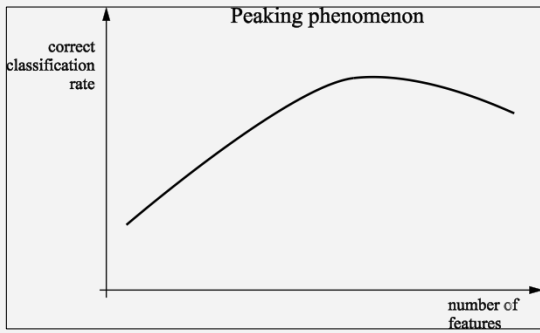
Goal of Feature Selection is to find the set of d features out of the set of D originally measured variables, where $d < D$ (if possible $d \ll D$), so as to maximize (or minimize) a chosen criterion.

23

Motivation for dimensionality reduction

- Reduce the complexity of classifiers
- Improve performance
 - Ugly duckling theorem (Watanabe)
 - Better generalization
 - Peaking phenomenon
- Reduce measurement extraction costs
- Assess class separability

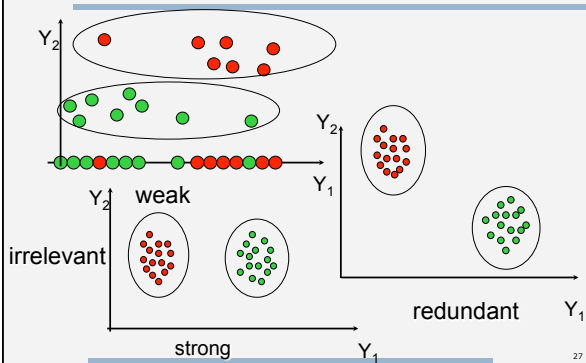
24



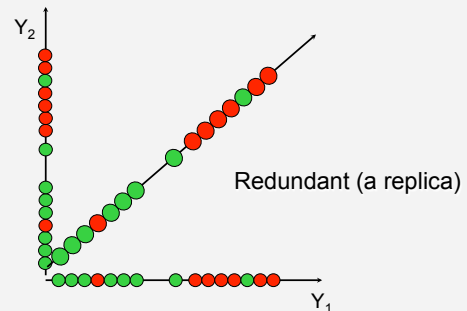
Which features to exclude?

- we exclude those measurements
 - which do not contain sufficient quantity of relevant information
 - or which are redundant
 - or completely irrelevant
- Remark:
 - redundancy does not mean that given feature (sattribute) has no information value !
 - redundancy is the consequence of strong statistical relations among variables

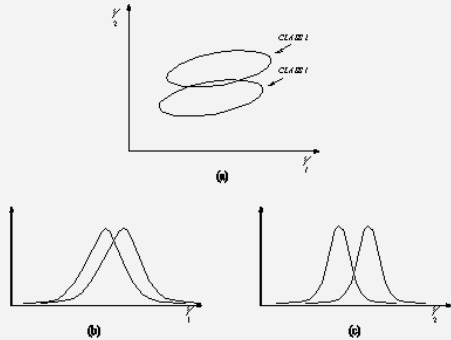
Irrelevant, strong, weak, and redundant features



Redundant features



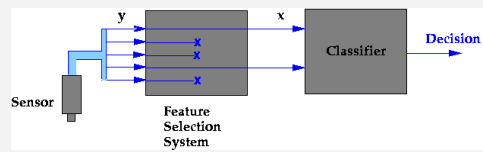
2D example of FS



29

Ingredients

- Feature selection criterion
- Feature set optimisation procedure
- Required dimensionality



30

FS Evaluation Criteria

- Error Probability
- Probabilistic Distance Measures
- Probabilistic Dependence Measures
- Entropy Measures
- Interclass Distance Measures

31

Example: 2-class separability measure

- Divergence J_D $J_D = \int [p(x|\omega_1) - p(x|\omega_2)] \log \frac{p(x|\omega_1)}{p(x|\omega_2)} dx$
- For instance, for normally distributed classes the divergence becomes
- Some class separability measures can be simplified analytically when the classes have parametric distributions $J_D = \frac{1}{2}(\mu_2 - \mu_1)^T [\Sigma_1^{-1} + \Sigma_2^{-1}] (\mu_2 - \mu_1) + \frac{1}{2} \text{tr} [\Sigma_1^{-1} \Sigma_2 + \Sigma_2^{-1} \Sigma_1 - 2I]$
- Mahalanobis distance $\Sigma_1 = \Sigma_2 = \Sigma$

$$J_M = (\mu_2 - \mu_1)^T \Sigma^{-1} (\mu_2 - \mu_1)$$

32

- **Approaches: Deterministic vs. Non-deterministic**
 - ✓ Deterministic
(all sequential algorithms)
 - ✓ Non-deterministic
Monte Carlo
Evolutionary approaches (genetic algorithms)
- **"Filter" vs. "Wrapper" approach**
 - "Filter" approach - evaluating probabilistic measures
 - "Wrapper" approach - evaluating classification rate

33

- Feature selection process modelled on the Indian Buffet metaphor
 - i-th customer (object) samples dishes (features) with a probability proportional to their popularity, and
 - samples a number of new dishes (features) defined by a prior



- N number of objects, and K a number of features
- Let $N \times K$ matrix Z

$$Z = \begin{bmatrix} z_{11} & \dots & z_{1K} \\ \vdots & & \vdots \\ z_{N1} & \dots & z_{NK} \end{bmatrix} \quad (10)$$

be a binary indicator matrix with $z_{ij} = 1$ denoting that feature j has been selected by object i

- The problem of feature selection can be formulated as

$$\max_Z J(X|Z) \quad (11)$$

where $J(X|Z)$ is a feature selection criterion

35

- Inference of Z by Markov Chain Monte Carlo sampling from posterior $P(Z|X)$ as

$$P(z_{ik}|X, Z_{-(ik)}) \propto J(X|Z)P(z_{ik} = 1|Z_{-(ik)}) \quad (12)$$

- $P(z_{ik} = 1|Z_{-(ik)})$ a prior controlling features selected

$$P(z_{ik}|X, Z_{-(ik)}) = \frac{\rho_{-(ik)}}{N} \quad (13)$$

with $\rho_{-(ik)}$ denoting the number of objects selecting feature k , and α is a meta parameter

- Gibbs sampling algorithm
 - Choose a binary matrix Z at random
 - For each column k , if $\rho_{-(ik)} > 0$, set $z_{ik} = 1$ with probability $\frac{\rho_{-(ik)}}{N}$. Otherwise, delete the column.
 - Add $\text{Poisson}(\frac{\alpha}{N})$ new columns with ones in the i -th row

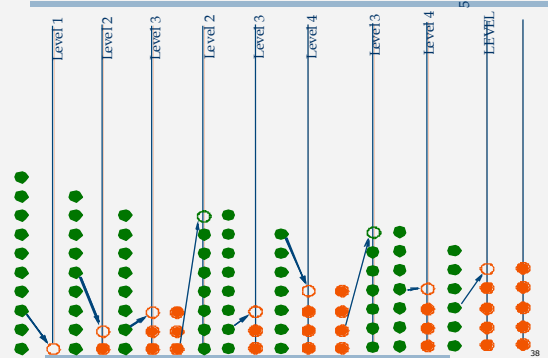
36

Sequential search algorithms

- Forward selection
 - Start from an empty set and select as first feature the individually best measurement
 - Assume k features have been selected. Then select from the remain $D-k$ measurements the one which in combination with the existing features gives the best set
- Backward selection
 - Start from a complete set of measurements
 - Remove one at a time (the least effective)
- Sequential search with backtracking

37

Plus l Take-Away r ($l=2, r=1$)



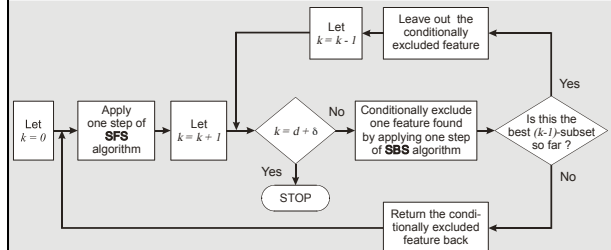
38

Floating search methods

- based on „backtracking“ in both directions
- resulting dimensionality in intermediate steps is not changing monotonously but is „floating“
- according to the prevailing search direction we have:
 - sequential backward floating search (SBFS)
 - sequential forward floating search (SFFS)

39

Scheme of SFFS algorithm



40

1. D Cai et al, SRDA: An efficient algorithm for large scale discriminant analysis, IEEE Trans Knowledge Engineering, Jan 2008.
2. CH Chan et al, Multiscale local phase quantisation for robust component-based face recognition using kernel fusion of multiple descriptors, IEEE PAMI, pp 1164-1177, 2013
3. F Yan et al, Non-Sparse Multiple Kernel Fisher Discriminant Analysis. Jnl Machine Learning Research, 607-642 2012
4. J Fan & J Lv, A selective overview of variable selection in high dimensional feature spaces, Statistica Sinica, pp 101-148, 2010
5. T L Griffiths & Z Ghahramani, The Indian buffet process: An introduction and review, Jnl. Machine Learning Research, pp 1185-1224, 2011
6. P Pudil et al, Floating search methods in feature selection, Pattern Recognition Letters, pp 1119-1125, 1994